Atelier B

XEmacs Interactive Prover Interface for Atelier B

User Manual

version V1.7.4

ATELIER B
XEmacs Interactive Prover Interface for Atelier B User Manual
version V1.7.4

Document made by CLEARSY.

This document is the property of CLEARSY and shall not be copied, duplicated or distributed, partially or totally, without prior written consent.

All products names are trademarks of their respective authors.

CLEARSY
Maintenance ATELIER B
Parc de la Duranne
320 avenue Archimède
Les Pléiades III - Bât. A
13857 Aix-en-Provence Cedex 3
France

Tel 33 (0)4 42 37 12 70
Fax 33 (0)4 42 37 12 71

mail: contact@atelierb.eu

Contents

1 Introduction **1**

2 Installation **3**

2.1 General Installation 3

3 Use **5**

3.1 Select a project 5

3.2 Select a component 5

3.3 Interactive proof 8

3.3.1 Toolbar 8

3.3.2 Status line 9

3.3.3 PRI menu 10

3.3.4 Interactive Proof Buffers 11

4 EmacsPRI is free software **19**

List of Figures

3.1 The *pri-project-list* buffer 6
3.2 The *pri-component-list* buffer 7
3.3 The **customize** buffer 11
3.4 The *pri-po-list* buffer 12
3.5 The *pri-log* buffer 14
3.6 The *pri-hypothesis-list* buffer 15
3.7 The *pri-proof-tree* buffer 17
3.8 The *pri-old-proof-tree* buffer 17

Chapter 1

Introduction

XEmacs Interactive Prover Interface (EmacsPRI) can be used with Atelier B 3.6 and 3.7. XEmacs v20.4 or v21.1 is required. As Atelier B, it works on Solaris or Linux systems. The following XEmacs packages are required: `base` and `efs`. XEmacs `text-modes` package is required for the EmacsPRI toolbar. If this package is not installed EmacsPRI works without toolbar.

XEmacs for Solaris can be downloaded from this internet site:

- http://sunfreeware.com

Chapter 2

Installation

This interface should be installed by each user, in the `~/emacs` directory of his own account. Thus each user can modify Lisp source code and adapt the interface to his needs. Lisp source code represents about 400 Kb.

2.1 General Installation

- go into `~/emacs` directory
- copy the `EMACSPRI.V1.7.4.tar.Z` file in `~/emacs` directory
- type in the following command:
  `zcat EMACSPRI.V1.7.4.tar.Z | tar xpf -`
- edit the `~/emacs/EMACSPRI/emacspri` file,
  set the following variables:
  - `ATELIERB_DIR`:
    should contain the full path to the directory that contains the `startBB` file (without the final "/").
    Ex: `"/home/atelierb/AB/bbin"` (for `/home/atelierb/AB/bbin/startBB`)
  - `ATELIERB_VERSION`:
    should contain the release of the Atelier B used: `"3.6"` or `"3.7"`
  - A particular resource must be set for EmacsPRI to work correctly. The resource and its value are:

    ```
    ATB*BATCH*Print_Messages_For_Emacs: TRUE
    ```

    This resource can be set for instance in the user resource file (`~/.AtelierB`). You just need to add the line above to the latter file.

Chapter 3

Use

Once installation is done, type in:

```
~/emacs/EMACSPRI/emacspri
```

The buffer which is displayed at startup then contains the project list.

3.1 Select a project

The project list is displayed in the `pri-projects-list` buffer (cf. figure 3.1 page 6). To open a project, just click with the middle button on the name of the project.
To exit, you can use either:

- the `Exit` button of the toolbar
- or the `File->Exit XEmacs` menu entry
- or the `C-x C-c` keyboard shortcut.

3.2 Select a component

Once a project is selected, its status is displayed in the `pri-components-list` buffer (cf. figure 3.2 page 7). To start the interactive proof of a component, just click with the middle button on the proof percentage of that component. Completely proved components can not be opened if proved proof obligations are not displayed (cf. paragraph 3.3.3).
To go back to the project list, use either:

- the `Exit` button of the toolbar
- or the `File->Exit XEmacs` menu entry
- or the `C-x C-c` keyboard shortcut.

‘

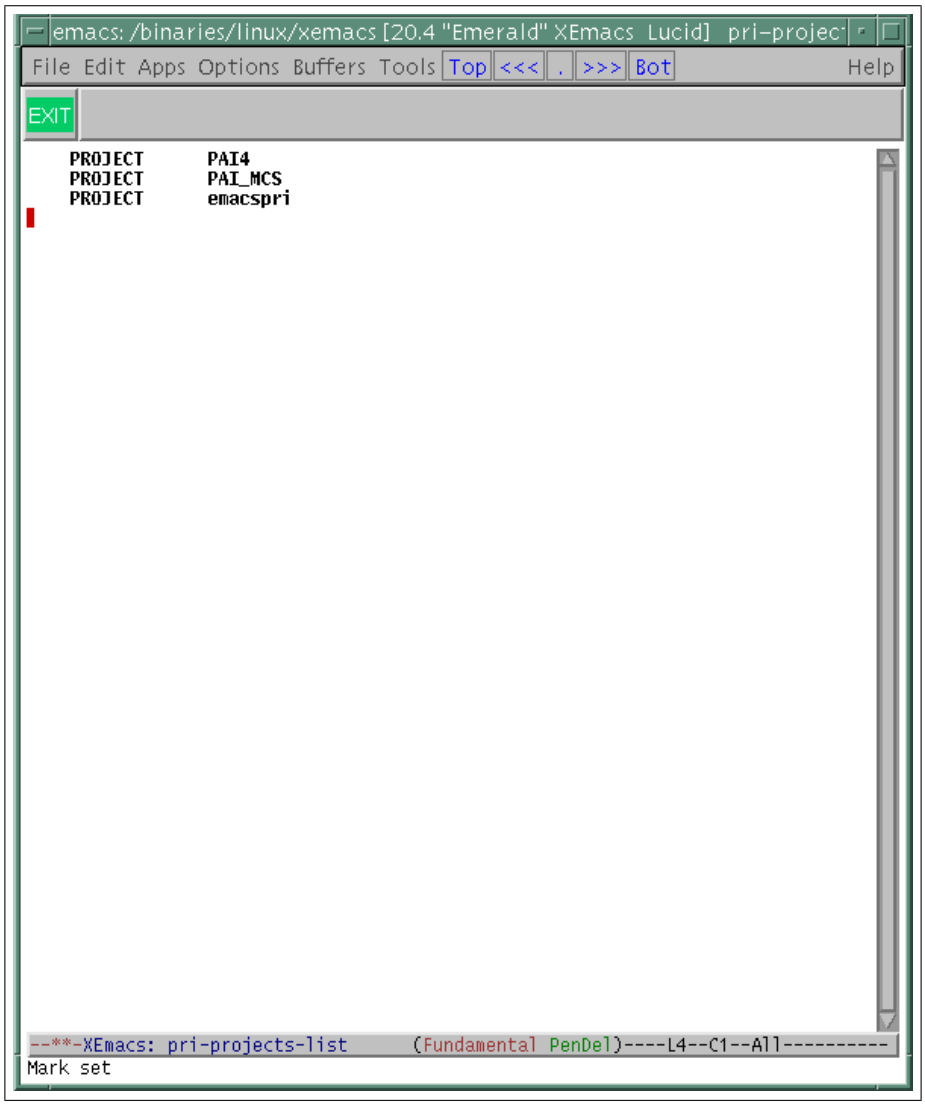
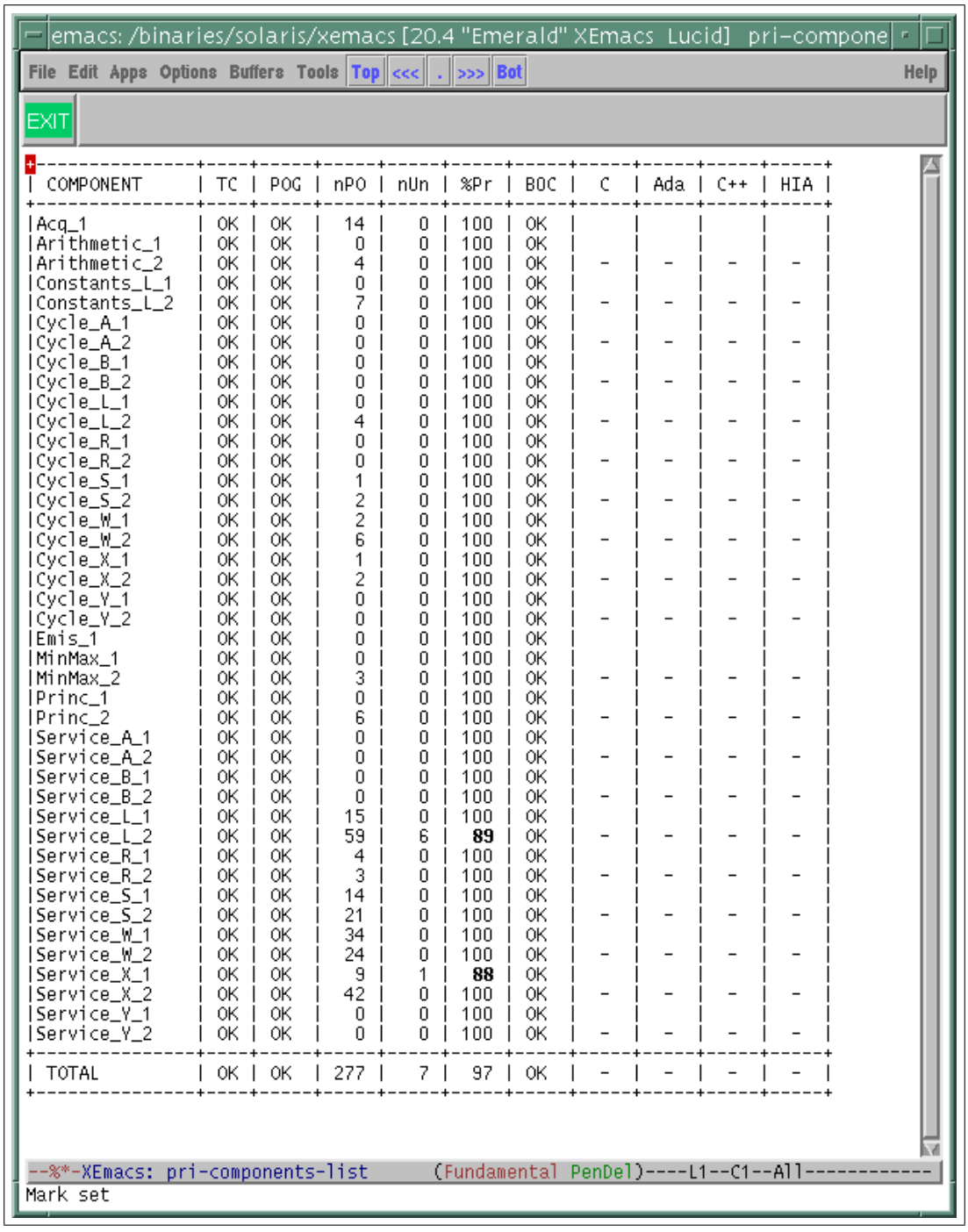


Figure 3.1: The *pri-project-list* buffer


emacs: /binaries/solaris/xemacs [20.4 "Emerald" XEmacs Lucid] pri-compone
File Edit Apps Options Buffers Tools Top <<< . >>> Bot Help
EXIT
--%*-XEmacs: pri-components-list (Fundamental PenDel)----L1--C1--All----------
Mark set

COMPONENT	TC	POG	nPO	nUn	%Pr	BOC	C	Ada	C++	HIA
Acq_1	OK	OK	14	0	100	OK				
Arithmetic_1	OK	OK	0	0	100	OK				
Arithmetic_2	OK	OK	4	0	100	OK	-	-	-	-
Constants_L_1	OK	OK	0	0	100	OK				
Constants_L_2	OK	OK	7	0	100	OK	-	-	-	-
Cycle_A_1	OK	OK	0	0	100	OK				
Cycle_A_2	OK	OK	0	0	100	OK	-	-	-	-
Cycle_B_1	OK	OK	0	0	100	OK				
Cycle_B_2	OK	OK	0	0	100	OK	-	-	-	-
Cycle_L_1	OK	OK	0	0	100	OK				
Cycle_L_2	OK	OK	4	0	100	OK	-	-	-	-
Cycle_R_1	OK	OK	0	0	100	OK				
Cycle_R_2	OK	OK	0	0	100	OK	-	-	-	-
Cycle_S_1	OK	OK	1	0	100	OK				
Cycle_S_2	OK	OK	2	0	100	OK	-	-	-	-
Cycle_W_1	OK	OK	2	0	100	OK				
Cycle_W_2	OK	OK	6	0	100	OK	-	-	-	-
Cycle_X_1	OK	OK	1	0	100	OK				
Cycle_X_2	OK	OK	2	0	100	OK	-	-	-	-
Cycle_Y_1	OK	OK	0	0	100	OK				
Cycle_Y_2	OK	OK	0	0	100	OK	-	-	-	-
Emis_1	OK	OK	0	0	100	OK				
MinMax_1	OK	OK	0	0	100	OK				
MinMax_2	OK	OK	3	0	100	OK	-	-	-	-
Princ_1	OK	OK	0	0	100	OK				
Princ_2	OK	OK	6	0	100	OK	-	-	-	-
Service_A_1	OK	OK	0	0	100	OK				
Service_A_2	OK	OK	0	0	100	OK	-	-	-	-
Service_B_1	OK	OK	0	0	100	OK				
Service_B_2	OK	OK	0	0	100	OK	-	-	-	-
Service_L_1	OK	OK	15	0	100	OK				
Service_L_2	OK	OK	59	6	**89**	OK	-	-	-	-
Service_R_1	OK	OK	4	0	100	OK				
Service_R_2	OK	OK	3	0	100	OK	-	-	-	-
Service_S_1	OK	OK	14	0	100	OK				
Service_S_2	OK	OK	21	0	100	OK	-	-	-	-
Service_W_1	OK	OK	34	0	100	OK				
Service_W_2	OK	OK	24	0	100	OK	-	-	-	-
Service_X_1	OK	OK	9	1	**88**	OK				
Service_X_2	OK	OK	42	0	100	OK	-	-	-	-
Service_Y_1	OK	OK	0	0	100	OK				
Service_Y_2	OK	OK	0	0	100	OK	-	-	-	-
TOTAL	OK	OK	277	7	97	OK	-	-	-	-

Figure 3.2: The *pri-component-list* buffer

3.3 Interactive proof

3.3.1 Toolbar

This toolbar is available as soon as a component has been selected (cf. figure 3.4 page 12). It contains buttons related to prover commands and buffer access. You must have installed the `text-mode` Lisp package and have your XEmacs compiled with support for `xpm`.

Right to left, the buttons displayed are:

- `Exit`
  End of interactive proof, returns to component list.
- `Disk`
  Save current proof (command `sw`).
- `te`
  Synonymous with `te - Try everywhere`
- `Refresh`
  Bring back the `PRI>` prompt when in case it has been lost.
- `Stop`
  Command interrupt.
- `Prev`
  Go to the previous Proof Obligation `pv - Previous PO`.
- `Next`
  Go to the next Proof Obligation `ne - Next`.
- `Back`
  Synonymous with `ba - Back`.
- `Step`
  Synonymous with `st - Step`.
- `dd`
  Synonymous with `dd - Deduction`.
- `dd0`
  Synonymous with `dd(0) - Deduction force 0`.
- `mp`
  Synonymous with `mp - Mini Proof`.
- `pp`
  Synonymous with `pp - Predicate Prover`.
- `pp0`
  Synonymous with `pp(rp.0) - Reduced Predicate Prover`.
- `pr`
  Synonymous with `pr - Prove`.

- `ss`
  Synonymous with `ss - Simplify Set`.
- `dd1`
  Synonymous with `dd(1) - Deduction force 1`.
- `dd2`
  Synonymous with `dd(2) - Deduction force 2`.
- `Old Proof`
  Display the buffer containing the old proof tree `pri-old-proof` in a new frame.
- `Proof Tree`
  Display the buffer containing the current proof tree `pri-proof-tree` in a new frame.
- `Hyp`
  Display the buffer containing the hypothesis stack `pri-hypothesis-list` in a new frame.
- `Term`
  Display the `pri-log` buffer (the one in which interactive proof is conducted) in a new frame.
- `Goal`
  Synonymous with `cg - Current Goal`.
- `PO List`
  Display the buffer containing the Proof Obligation list `pri-po-list` in a new frame.

3.3.2 Status line

The Term and PO list buffers contain at their bottom a status line (cf. figure 3.4 page 12), showing:

1. Number of proved PO
2. Total number of PO
3. Status (Proved/Unproved) of the current PO
4. Current component
5. Current operation
6. Current PO

The displayed format is:
1Un/2PO 3 4.5.6

ex: 3Un/15PO Unproved composant1.op1.1

3.3.3 PRI menu

It contains the following items:

- `About Emacspri...`

  Horizontally split the current window and open a new buffer `*About Emacspri..*` in the bottom frame. This buffer contains:

  - the EmacsPRI version number ;
  - the directory, version number and options of the underlying Atelier B used ;
  - host name.

  Example:

```
EMACSPRI : version V1.7.4

ATELIERB :
        Directory : /beta/3.7Beta4/AB/bbin
        Version : 3.7
        Option : extra_large

HOST :
        cubitus
```

- `Save`

  Synonymous with `sw - Save`

- `Customize`

  Open the Interface Customize buffer: `*Customize Group : Pri*` (cf. figure 3.3 page 11). This buffer contains editable fields, allowing to modify the behaviour and the apparence of the interface.

  The first line is composed of the name of the variable and of the possible values for that variable. The `[Toggle]` button modifies the value of the variable (switch between `off` and `on`, selection with middle button). The second line shows the current value of this variable. The third line gives a short description of the use of that variable.

  After having modified some variable, click on the `set` button to apply your changes.

  The `save` button saves the value of that variable in your `.emacs` file.

  Variables that can be customized are:

  - `Show Proved Pos` Indicates whether Proved PO should be displayed.
  - `Inhibit Toolbar` Indicates whether the toolbar should be displayed.
  - `Hypothesis Buffer` Indicates whether the hypothesis stack should be displayed.
  - `Trace` Indicates whether traces should be displayed in the `*scratch*` buffer.

- `Exit C-x C-c` Exit Interactive Proof Mode.

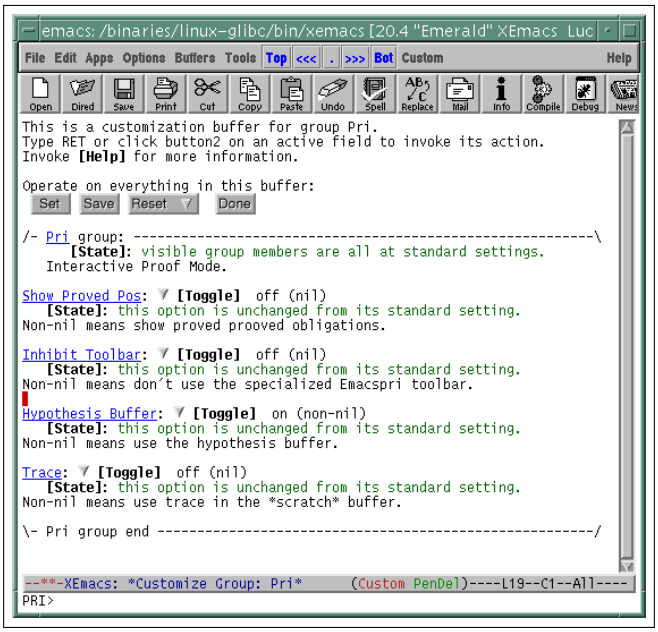

emacs: /binaries/linux-glibc/bin/xemacs [20.4 "Emerald" XEmacs Luc
File Edit Apps Options Buffers Tools Top <<< . >>> Bot Custom Help
Open Dired Save Print Cut Copy Paste Undo Spell Replace Mail Info Compile Debug News
This is a customization buffer for group Pri.
Type RET or click button2 on an active field to invoke its action.
Invoke [Help] for more information.
Operate on everything in this buffer:
Set Save Reset Done
/- Pri group: ----------\
[State]: visible group members are all at standard settings.
Interactive Proof Mode.
Show Proved Pos: [Toggle] off (nil)
[State]: this option is unchanged from its standard setting.
Non-nil means show proved prooved obligations.
Inhibit Toolbar: [Toggle] off (nil)
[State]: this option is unchanged from its standard setting.
Non-nil means don't use the specialized Emacspri toolbar.
Hypothesis Buffer: [Toggle] on (non-nil)
[State]: this option is unchanged from its standard setting.
Non-nil means use the hypothesis buffer.
Trace: [Toggle] off (nil)
[State]: this option is unchanged from its standard setting.
Non-nil means use trace in the *scratch* buffer.
\- Pri group end ----------/
--**-XEmacs: *Customize Group: Pri* (Custom PenDel)----L19--C1--All----
PRI>

Figure 3.3: The **customize** buffer

3.3.4 Interactive Proof Buffers

Proof Obligation List: buffer pri-po-list

This buffer contains the list of the unproved PO of the selected component (cf. figure 3.4 page 12). The default behaviour is that only unproved PO are displayed. To also display proved PO, you have to customize the interface with `PRI->Customize` (cf. paragraph 3.3.3).

To go on a particular PO, click (middle button) on the name of the PO, the `pri-log` buffer is then displayed.

Proof Terminal: buffer pri-log

This buffer contains traces of past commands as well as the current goal (cf. figure 3.5 page 14), using the format:

```
cmd:<command name>
<prover messages>
goal:
<current goal>
```

A green underlined message `PO Proved` is displayed when the current PO has just been proved. This buffer can't be edited.

The mini-buffer at the bottom of the XEmacs window allows to type in commands when the `PRI>` prompt is displayed. Keyboard arrows (up and down) allow to browse the command history.

A pop-up menu can be displayed by a right click. It contains the following items:

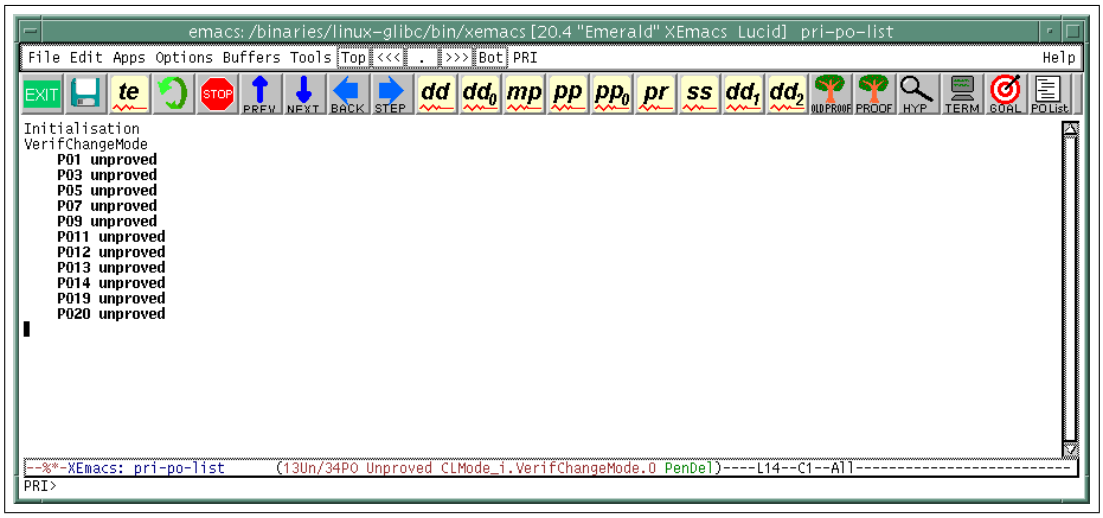


Figure 3.4: The *pri-po-list* buffer

- ah - Add hypothesis
- eh(Goal) - Use Equality in Hypothesis in Goal
- sh - Search Hypothesis
- dc - Do Case
- dcs - Do Case Special
- fh - False hypothesis
- mh - Modus Ponens
- ae - Abstract Expression (Expr = pri_varN)
- se - Suggest For Exist
- ----------
- st(End) - Step to End
- re - Reset
- ----------
- ap - Arithmetic Proof
- ct - Contradiction
- cts - Special Contradiction
- tp(Hyp) - Proof by attempts (based on hypothesis)
- tp(Goal) - Proof by attempts (based on goal)

- ----------
- ah & pr
- ah & mp
- ah & pp(rp.0)
- mh & pr
- mh & mp
- dc & pr
- dc & mp
- dd & eh
- dd & mh
- ba & dd

Selecting one of these items executes the associated command(s).

To execute ah, eh, sh, dc, dcs, mh, ae or se commands, the user should first select an expression in the buffer, with the left button, then select the command to execute, with the right button.

The last entries execute two commands at once. To execute some of these commands, the user should first select an expression or an hypothesis in the buffer.

Hypotheses stack: buffer pri-hypothesis

This buffer is not displayed by default. To display it, you have to customize the interface with PRI->Customize menu (cf. paragraph 3.3.3).

This buffer contains the hypothesis stack (cf. figure 3.6, page 15). Each hypothesis is equipped with a square button. A left click on one hypothesis selects it. An other click deselects it.

A pop-up menu can be displayed, with the right button. It contains the following items:

- ah - Add hypothesis
- fh - False hypothesis
- eh(Goal) - Use Equality in Hypothesis in Goal
- eh(AllHyp) - Use Equality in Hypothesis in All Hypothesis
- sh - Search Hypothesis
- mh - Modus Ponens

Selecting one ot these items executes the associated command, in relation with the selected hypotheses.

To execute a sh command on an expression, this expression should first be selected in the buffer, with the left button, then the sh command is executed, with the right button.

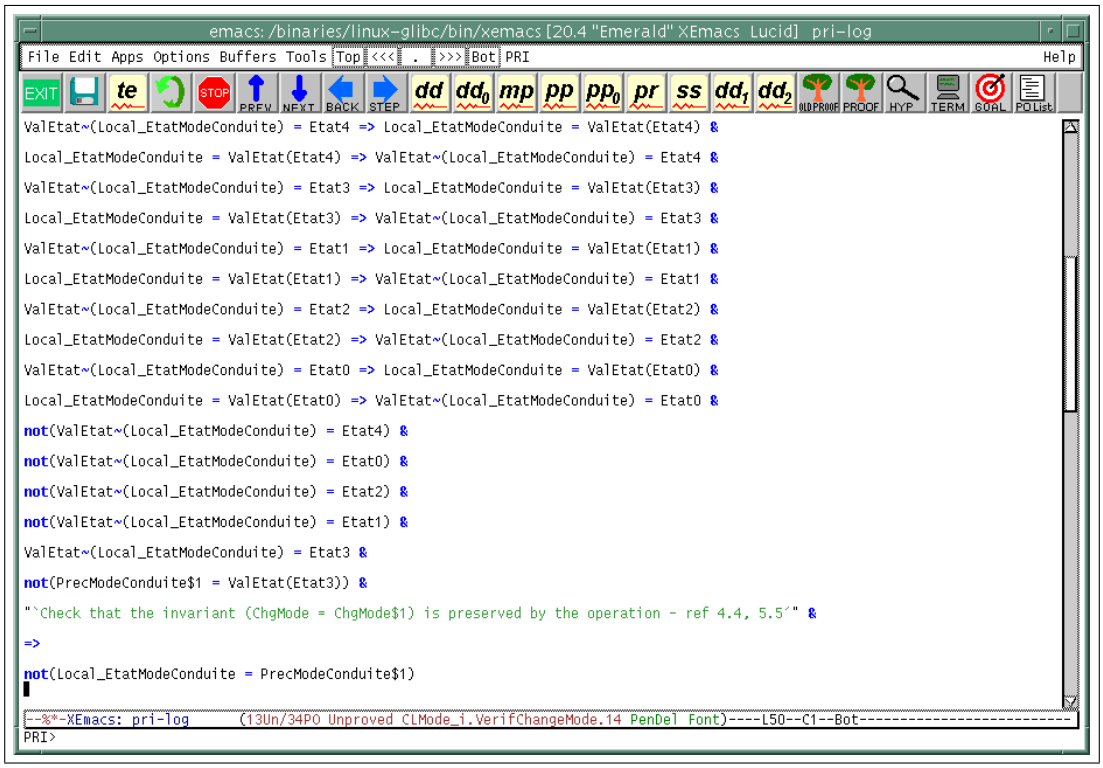

emacs: /binaries/linux-glibc/bin/xemacs [20.4 "Emerald" XEmacs Lucid] pri-log
File Edit Apps Options Buffers Tools Top <<< . >>> Bot PRI Help
EXIT te STOP PREV NEXT BACK STEP dd dd0 mp pp pp0 pr ss dd1 dd2 AUTPROOF PROOF HYP TERM GOAL POList
ValEtat~(Local_EtatModeConduite) = Etat4 => Local_EtatModeConduite = ValEtat(Etat4) &
Local_EtatModeConduite = ValEtat(Etat4) => ValEtat~(Local_EtatModeConduite) = Etat4 &
ValEtat~(Local_EtatModeConduite) = Etat3 => Local_EtatModeConduite = ValEtat(Etat3) &
Local_EtatModeConduite = ValEtat(Etat3) => ValEtat~(Local_EtatModeConduite) = Etat3 &
ValEtat~(Local_EtatModeConduite) = Etat1 => Local_EtatModeConduite = ValEtat(Etat1) &
Local_EtatModeConduite = ValEtat(Etat1) => ValEtat~(Local_EtatModeConduite) = Etat1 &
ValEtat~(Local_EtatModeConduite) = Etat2 => Local_EtatModeConduite = ValEtat(Etat2) &
Local_EtatModeConduite = ValEtat(Etat2) => ValEtat~(Local_EtatModeConduite) = Etat2 &
ValEtat~(Local_EtatModeConduite) = Etat0 => Local_EtatModeConduite = ValEtat(Etat0) &
Local_EtatModeConduite = ValEtat(Etat0) => ValEtat~(Local_EtatModeConduite) = Etat0 &
not(ValEtat~(Local_EtatModeConduite) = Etat4) &
not(ValEtat~(Local_EtatModeConduite) = Etat0) &
not(ValEtat~(Local_EtatModeConduite) = Etat2) &
not(ValEtat~(Local_EtatModeConduite) = Etat1) &
ValEtat~(Local_EtatModeConduite) = Etat3 &
not(PrecModeConduite$1 = ValEtat(Etat3)) &
"`Check that the invariant (ChgMode = ChgMode$1) is preserved by the operation - ref 4.4, 5.5´" &
=>
not(Local_EtatModeConduite = PrecModeConduite$1)
--%*-XEmacs: pri-log (13Un/34PO Unproved CLMode_i.VerifChangeMode.14 PenDel Font)----L50--C1--Bot----------
PRI>

Figure 3.5: The *pri-log* buffer

Current proof tree: buffer pri-proof-tree

This buffer contains the current proof tree (cf. figure 3.7, page 17). By default, only unterminated branches are displayed. Once a branch is completed, it is automatically hidden except the first command which is abbreviated and completed with an ellipsis.

This buffer is equipped with the following scrolldown menu:

- `Headings`

  This menu allow to navigate into the proof tree. A Heading corresponds to a proof command:

  - `Up`
    Go up.
  - `Next`
    Go to the next command.
  - `Previous`
    Go to the previous command.
  - `Next Same Level`
    Go to the next command at the same level.
  - `Previous Same Level`
    Go to the previous command at the same level.

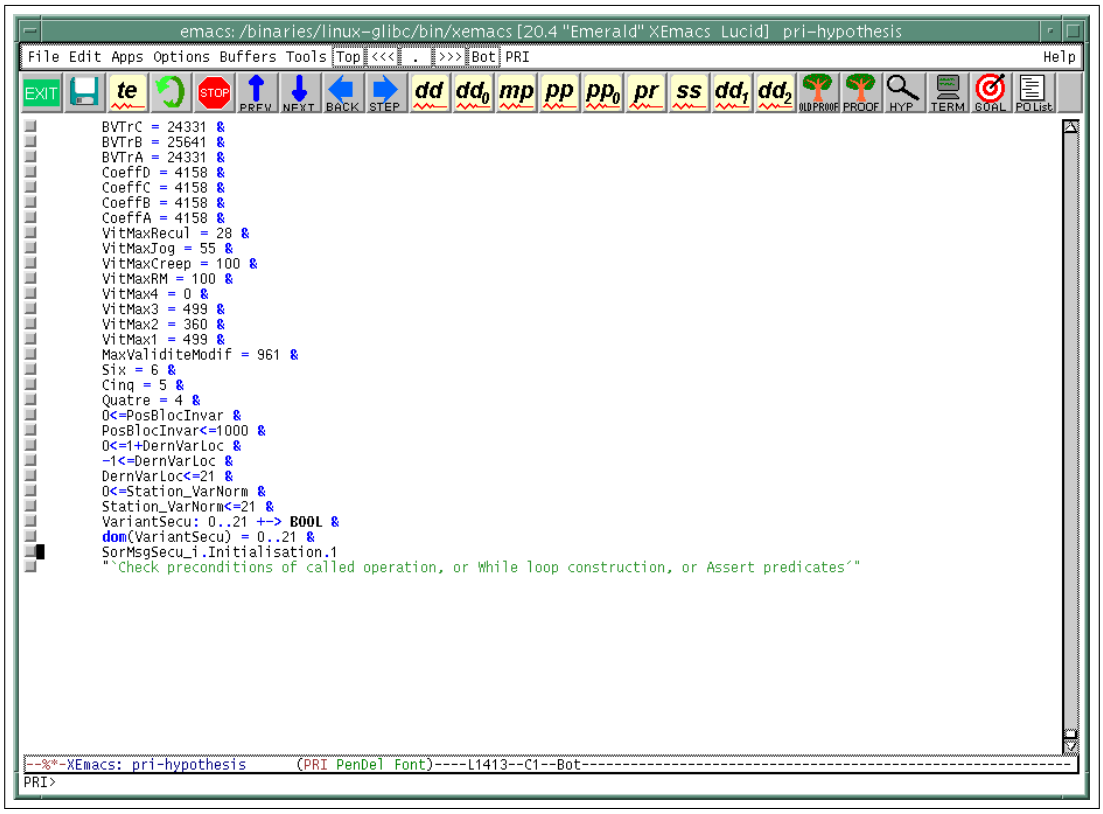

emacs: /binaries/linux-glibc/bin/xemacs [20.4 "Emerald" XEmacs Lucid] pri-hypothesis
File Edit Apps Options Buffers Tools Top <<< . >>> Bot PRI Help
EXIT te STOP PREV NEXT BACK STEP dd dd0 mp pp pp0 pr ss dd1 dd2 OLDPROOF PROOF HYP TERM GOAL POList
BVTrC = 24331 &
BVTrB = 25641 &
BVTrA = 24331 &
CoeffD = 4158 &
CoeffC = 4158 &
CoeffB = 4158 &
CoeffA = 4158 &
VitMaxRecul = 28 &
VitMaxJog = 55 &
VitMaxCreep = 100 &
VitMaxRM = 100 &
VitMax4 = 0 &
VitMax3 = 499 &
VitMax2 = 360 &
VitMax1 = 499 &
MaxValiditeModif = 961 &
Six = 6 &
Cinq = 5 &
Quatre = 4 &
0<=PosBlocInvar &
PosBlocInvar<=1000 &
0<=1+DernVarLoc &
-1<=DernVarLoc &
DernVarLoc<=21 &
0<=Station_VarNorm &
Station_VarNorm<=21 &
VariantSecu: 0..21 +-> BOOL &
dom(VariantSecu) = 0..21 &
SorMsgSecu_i.Initialisation.1
"`Check preconditions of called operation, or While loop construction, or Assert predicates´"
-:%*-XEmacs: pri-hypothesis (PRI PenDel Font)---L1413--C1--Bot
PRI>

Figure 3.6: The *pri-hypothesis-list* buffer

- `Show`

  This menu allows to show commands:

  - `Show All`
    Show all commands.
  - `Show Entry`
    Show the current selection.
  - `Show Branches`
    Show hidden branches of the selection
  - `Show Children`
    Show hidden branches of the next level only
  - `Show Subtree`
    Show all hidden branches of the current selection.

- `Hide`

  This menu allows to hide commands:

  - `Hide Leaves`
    This entry is a predefined XEmacs menu entry. This item is not used by EmacsPRI.
  - `Hide Body`
    This entry is a predefined XEmacs menu entry. This item is not used by EmacsPRI.
  - `Hide Entry`
    This entry is a predefined XEmacs menu netry. This item is not used by EmacsPRI.
  - `Hide Subtree`
    Hide the commands below the current command
  - `Hide Others`
    Hide all the commands except the current command and upper level commands.
  - `Hide Sublevels`
    Hide lower level commands

A pop-up menu can be displayed, with the right button. It contains the following items:

- `ba - Back`
- `st - Step`
- `st(End) - Step to End`
- `re - Reset`
- `Proof Paste`

For the `Proof Paste` menu entry, a portion of the proof tree should be first selected. Then, this command executes the list of selected commands.

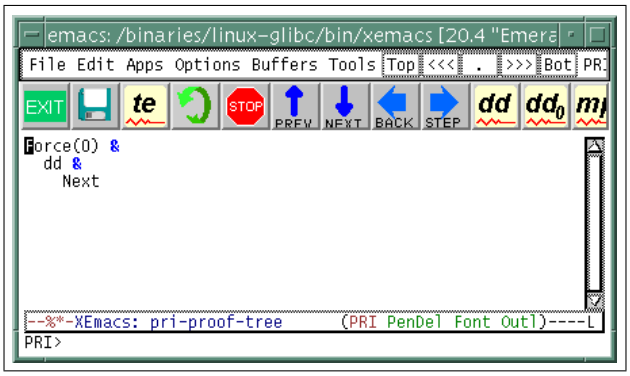


Figure 3.7: The *pri-proof-tree* buffer

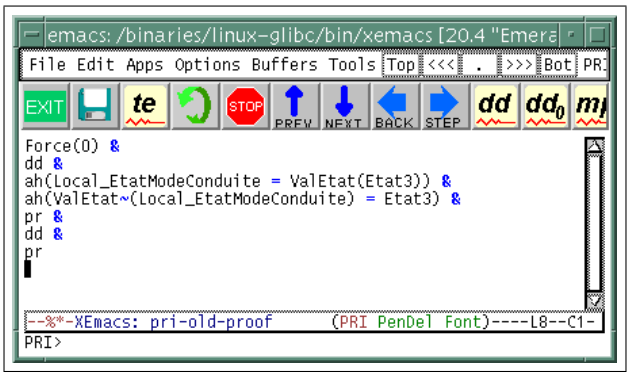


Figure 3.8: The *pri-old-proof-tree* buffer

Old proof tree: buffer pri-old-proof

This buffer contains the old proof tree (cf. figure 3.8 page 17). The pop-up menu is the same as in the current proof tree buffer `pri-proof-tree`.

Chapter 4

EmacsPRI is free software

This interface is free software. It is distributed by ClearSy. It is an experimental version, so it is distributed without any warranty. Lisp source code is distributed with the interface. You can modify source code for your own use, but you can't modify it to commercialize it. ClearSy can help you to realise modifications. ClearSy is interested in significant functional development you can make on this interface. ClearSy can possibly integrate it into future versions.

ClearSy makes available new versions of this interface on its internet site:

`http://www.atelierb.societe.com/emacspri/emacspri_uk.html`